

## Some Examples for Solving Systems of Algebraic Equations by Calculating Groebner Bases

W. BOEGE, R. GEBAUER AND H. KREDEL

*University of Heidelberg, Institute for Applied Mathematics,  
Heidelberg, F.R.G.*

(Received 14 November 1984)

---

Some notes on the computation of Groebner Bases using Buchberger's algorithm over different coefficient domains and the solution of systems of algebraic equations using Groebner bases are given. Examples demonstrate applicability and its current limits, and show the wide range of problems tractable by this method. The choices of an appropriate variable ordering and a suitable term ordering are of crucial importance for the computing time and space allocation. The "optimal" variable ordering is considered, and an improvement for the selection of valid solutions of the system of algebraic equations is described.

---

### 1. Introduction

In different fields of applications it is sometimes required that we determine the, finitely, many solutions of a system of algebraic equations in several variables with rational or integral coefficients, or otherwise get the answer that there are infinitely many solutions. Sometimes it is also of theoretical interest whether a special coordinate of a solution is exactly equal to  $\sqrt{2}$  or  $2/3$  for example. Neither this nor the warranty that all solutions are found, can be achieved by numerical iterative approximation methods.

### 2. Method

The solutions of systems of algebraic equations can be constructed using the properties of Groebner Bases.

First, the set of polynomials  $F$  that defines the system of algebraic equations is transformed into a Groebner Basis  $G$ , which is a set of polynomials satisfying  $\text{ideal}(F) = \text{ideal}(G)$  and thus has the same set of solutions as  $F$ . The theory of Groebner Bases has been initiated by Buchberger (1965, 1970). For a concise introduction see (Buchberger, 1985).

Now, in the case where only finitely many solutions exist, Method 6.12. of Buchberger (1985), introduced in Buchberger (1965), can be applied, which is based on the construction of univariate polynomials for each variable (by solving systems of linear equations in the residue class ring modulo the given ideal). The result is a finite inclusion for the solutions of the system of algebraic equations.

In the case of rational coefficients for every univariate polynomial the isolating intervals for the real roots can be computed by known methods, for example by the method of Uspenski, using the continued fraction approach, or by Sturm sequences and repeated interval bisection, see Collins & Loos (1982).

Not all combinations of the real roots of the univariate polynomials for each variable are solutions of the system of algebraic equations. The set of the combinations that are solutions can be restricted by:

#### *Factorisation of the univariate polynomials*

The finite inclusion for the real and complex solutions can be refined by the decomposition of the ideal generated by the system of algebraic equations. For that purpose the univariate polynomials can be factorised (Kaltofen, 1982) and adjoined to the ideal (Schrader, 1976). This yields an improvement of the last step in the method described in Buchberger (1965, 1985):

*Algorithm:*  $L \leftarrow \text{DIRGZD}(k, G)$

[Zero dimensional ideal decomposition.]

*Input:* An index  $k$ ,  $1 \leq k \leq r$ , indicating the variable, for which the univariate polynomial should be constructed.

$G$  = Groebner Basis of the system of algebraic equations in  $r$  variables,  $1 \leq r$ .

*Output:*

$L = (L_1, \dots, L_s)$  a list of Groebner Bases such that:

- (a)  $\text{SOL}(G) = \text{SOL}(L_1) \cup \dots \cup \text{SOL}(L_s)$
- (b)  $L_i$  contains a univariate polynomial in the variable  $x_j$   
(for all  $1 \leq i \leq s$  and for all  $k \leq j \leq r$ )
- (c)  $\text{ideal}(L_i) \cup \text{ideal}(L_j) = \text{ideal}(1)$   
(for all  $1 \leq i, j \leq s$ ,  $i \neq j$ ).

(1) [initialise.]  $L \leftarrow ()$ .  $r \leftarrow$  number of variables.

(2) [case  $k < r$  (recursion) and  $k = r$ .]

$p \leftarrow \text{DIRMPG}(k, G)$ .

[determine the univariate polynomial  $p$  in  $x_k$  of lowest degree in  $\text{ideal}(G)$  by Method 6.11 in Buchberger (1985)].

$p = p_1^{e(1)} \dots p_l^{e(l)}$ . [factorisation.]

for  $n = 1, \dots, l$  do {

$F \leftarrow \text{Groebner Basis}(G \cup \{p_n\})$ .

if  $k = r$  then  $L' \leftarrow (F)$

else  $L' \leftarrow \text{DIRGZD}(k+1, F)$ .

$L \leftarrow L \sqcup L'$  [concatenation]}.

#### *Optimisation of the variable ordering*

In most applications the computation time for a Groebner Basis is strongly dependent on the chosen variable ordering and term ordering. (See the example from Trinks (1978) in "summary of computing times".) Possible choices are the "inverse lexicographical" ("purely lexicographical") or the "inverse graduated" ("total degree") term ordering. See Buchberger (1985) and Trinks (1978) for the definitions and properties of these orderings.

To heuristically find an "optimal" variable ordering one looks at the "reduced univariate polynomials":

The “reduced univariate polynomial” corresponding to a polynomial

$$f(x_1, \dots, x_r) \in K[x_1, \dots, x_r]$$

for the variable  $x_i (1 \leq i \leq r)$  is defined by:

$$p_i(x_i) = g(1, \dots, 1, x_i, 1, \dots, 1) \in \mathbb{N}[x_i]$$

with  $g = \sum x^{(i)}$  when  $f = \sum a_{(i)} x^{(i)}$ ,  $a_{(i)} \neq 0$ .

The reduced polynomial for a set of polynomials is the sum over all reduced polynomials corresponding to the elements of the set.

Tests for computing times of Groebner Bases and factorising of multivariate polynomials have shown:

The variable ordering is heuristically “optimal”

if  $p_1(x) \geq \dots \geq p_r(x)$ ,

where the univariate polynomials are ordered according to the following ordering:

$h(x) > 0 \Leftrightarrow$  the leading coefficient of  $h$  is  $> 0$ ,

$h(x) > k(x) \Leftrightarrow h(x) - k(x) > 0$ .

However, there are some special examples where another variable ordering is better. The computation of the “reduced univariate polynomials” and the reordering of the variables is not very time consuming.

*The overall structure of the method implemented*

1. Find the “optimal” variable ordering according to the above description.
2. Compute the Groebner Basis by Buchberger’s algorithm.
3. Compute the univariate polynomials and decompose the given ideal by factorisation of the univariate polynomials by program DIRGZD.
4. Find the isolating intervals for the real roots of the univariate polynomials.
5. In a last step the user of this package can choose those combinations of (real) roots of the univariate polynomials he accepts as solutions of his original system. (As soon as, in step 4, numerical solutions are produced, the selection of combinations depends on a numerical criterion defining which tuples of values are accepted as solutions.)

*Notes regarding the implementation*

- The implementation is on the basis of the SAC-2 computer algebra system, which includes for example a (univariate) factorisation package and a real root isolation package for univariate polynomials (Collins & Loos, 1980). Recently an implementation in REDUCE 3.2 (Hearn, 1985) has been undertaken by R. Gebauer and A. C. Hearn. For this a new polynomial representation was introduced in the REDUCE 3.2 system by R. Gebauer, A. C. Hearn and H. Kredel.
- The distributive polynomial representation is more suitable for computation in polynomial ideal theory than the recursive representation. Especially for the computation of Groebner Bases, rapid access to the power products and more freedom for ordering the power products is needed. An appropriate package is found in Gebauer & Kredel (1983a).
- Our implementation of Buchberger’s algorithm is based on the implementation described in Winkler *et al.* (1981) for SAC-1 with several technical changes (Gebauer & Kredel, 1983b).

### 3. Examples

The examples are listed in the way they are input to the DIPIOS (*D*istributive *P*olynomial *I*nterface/*O*utput System) program:

D = coefficient domain

Q = rational numbers

MP = integers modulo p

F(.) = rational functions in the variables listed

R = D(X1, X2, ...) where X1, X2, ... are variables of the polynomial ring.

OPT = options

I = information about intermediate results

O = optimisation of the variable ordering by preprocessor

G = inverse graduated term ordering

L = inverse lexicographical term ordering (default)

P = determine the univariate polynomials

En = precision of n decimal digits for the isolating intervals for the real roots

In general, for first experiments, the following options should be used:

D = Q

R = D(X1, X2, ...)

OPT = ILO P E10

*Series of examples of algebraic equations occurring in the construction of Runge-Kutta methods for solving differential equations, see for instance Stoer & Bulirsch (1979) and Henrici (1961)*

The following five examples have been communicated by Hairer (personal communication). We show the input polynomials and make brief comments on the computing times.

#### 1. 6 polynomials in 8 variables

\$(HAIRER, RUNGE-KUTTA 1, 05.11.83)

D = Q

R = D(C2, C3, B3, B2, B1, A21, A32, A31)

OPT = IO

(

(+ C2 - A21)

(+ C3 - A31 - A32)

(+ B1 + B2 + B3 - 1)

(+ B2 C2 + B3 C3 - 1/2)

(+ B2 C2\*\*2 + B3 C3\*\*2 - 1/3)

(+ B3 A32 C2 - 1/6)

)

Computing time for Groebner Basis on IBM 3081D: 482 ms

*Note:* Infinitely many solutions, C2 and C3 are parameters.

## 2. 11 polynomials in 13 variables

\$(HAIRER, RUNGE-KUTTA 2, 05.11.1983)

D = Q

R = D(C2, C3, C4, B4, B3, B2, B1, A21, A31, A32, A41, A42, A43)

OPT = IO

```
(
(+ B1 + B2 + B3 + B4 - 1)
(+ B2 C2 + B3 C3 + B4 C4 - 1/2)
(+ B2 C2**2 + B3 C3**2 + B4 C4**2 - 1/3)
(+ B3 A32 C2 + B4 A42 C2 + B4 A43 C3 - 1/6)
(+ B2 C2**3 + B3 C3**3 + B4 C4**3 - 1/4)
(+ B3 C3 A32 C2 + B4 C4 A42 C2 + B4 C4 A43 C3
- 1/8)
(+ B3 A32 C2**2 + B4 A42 C2**2 + B4 A43 C3**2
- 1/12)
(+ B4 A43 A32 C2 - 1/24)
(+ C2 - A21)
(+ C3 - A31 - A32)
(+ C4 - A41 - A42 - A43)
)
```

Computing time for Groebner Basis on IBM 3081D:

```
367 676 ms (to find (C4-1) in the ideal)
+ 47 882 ms (new start with (C4-1) included)
= 415 558 ms (total)
```

*Note:* Infinitely many solutions, C2 and C3 are parameters.

## 3. 16 polynomials in 14 variables

\$(HAIRER, RUNGE-KUTTA 3, 10.11.1983)

D = Q

R = D(C2, C3, C4, C5, B2, B3, B4, B5, A32, A42, A43, A52, A53, A54)

OPT = GIO

```
(
(+ B2 C2 + B3 C3 + B4 C4 + B5 C5 - 1/2)
(+ B2 C2**2 + B3 C3**2 + B4 C4**2 + B5 C5**2
- 1/3)
(+ B3 A32 C2 + B4 A42 C2 + B4 A43 C3 + B5 A52 C2
+ B5 A53 C3 + B5 A54 C4 - 1/6)
(+ B2 C2**3 + B3 C3**3 + B4 C4**3 + B5 C5**3
- 1/4)
(+ B3 C3 A32 C2 + B4 C4 A42 C2 + B4 C4 A43 C3
+ B5 C5 A52 C2
+ B5 C5 A53 C3 + B5 C5 A54 C4 - 1/8)
(+ B3 A32 C2**2 + B4 A42 C2**2 + B4 A43 C3**2
+ B5 A52 C2**2
+ B5 A53 C3**2 + B5 A54 C4**2 - 1/12)
(+ B4 A43 A32 C2 + B5 A53 A32 C2 + B5 A54 A42 C2
```

```

+ B5 A54 A43 C3 - 1/24)
(+ B2 C2**4 + B3 C3**4 + B4 C4**4 + B5 C5**4 - 1/5)
(+ B3 C3**2 A32 C2 + B4 C4**2 A42 C2 + B4 C4**2 A43 C3
+ B5 C5**2 A52 C2
+ B5 C5**2 A53 C3 + B5 C5**2 A54 C4 - 1/10)
(+ B3 C2**2 A32 C3 + B4 C2**2 A42 C4 + B4 C3**2 A43 C4
+ B5 C2**2 A52 C2
+ B5 C3**2 A53 C5 + B5 C4**2 A54 C5 - 1/15)
(+ B4 C4 A43 A32 C2 + B5 C5 A53 A32 C2 + B5 C5 A54 A42 C2
+ B5 C5 A54 A43 C3 - 1/30)
(+ B3 A32**2 C2**2 + B4 A42**2 C2**2 + 2 B4 A42 C2 A43 C3
+ B4 A43**2 C3**2 + B5 A52**2 C2**2 + B5 A53**2 C3**2
+ B5 A54**2 C4**2
+ 2 B5 A52 C2 A53 C3 + 2 B5 A52 C2 A54 C4
+ 2 B5 A53 C3 A54 C4 - 1/20)
(+ B3 A32 C2**3 + B4 A42 C2**3 + B4 A43 C3**3
+ B5 A52 C2**3
+ B5 A53 C3**3 + B5 A54 C4**3 - 1/20)
(+ B4 A43 C3 A32 C2 + B5 A53 C3 A32 C2 + B5 A54 C4 A42 C2
+ B5 A54 C4 A43 C3 - 1/40)
(+ B4 A43 A32 C2**2 + B5 A53 A32 C2**2 + B5 A54 A42 C2**2
+ B5 A54 A43 C3**2 - 1/60)
(+ B5 A54 A43 A32 C2 - 1/120)
)

```

Computing time for Groebner Basis on IBM 3081D: 99 685 ms

*Note:* No solution, i.e. 1  $\notin$  Groebner Basis.

#### 4. 16 polynomials in 20 variables

```

$(HAIRER, RUNGE-KUTTA 4, P = 5 S = 6, 20.12.1983)
D = Q
R = D(C2, C3, C4, C5, C6, B2, B3, B4, B5, B6, A32, A42, A43, A52, A53,
A54, A62, A63, A64, A65)
OPT = OIL
(
(+ B2 C2 + B3 C3 + B4 C4 + B5 C5 + B6 C6 - 1/2)
(+ B2 C2**2 + B3 C3**2 + B4 C4**2 + B5 C5**2 + B6 C6**2
- 1/3)
(+ B3 A32 C2 + B4 A42 C2 + B4 A43 C3 + B5 A52 C2
+ B5 A53 C3
+ B6 A62 C2 + B6 A63 C3 + B6 A64 C4 + B6 A65 C5
+ B5 A54 C4 - 1/6)
(+ B2 C2**3 + B3 C3**3 + B4 C4**3 + B5 C5**3 + B6 C6**3
- 1/4)
(+ B3 C3 A32 C2 + B4 C4 A42 C2 + B4 C4 A43 C3
+ B5 C5 A52 C2
+ B6 C6 A62 C2 + B6 C6 A63 C3 + B6 C6 A64 C4
+ B6 C6 A65 C5

```

$$\begin{aligned}
& + B5 C5 A53 C3 + B5 C5 A54 C4 - 1/8) \\
& (+ B3 A32 C2^{**2} + B4 A42 C2^{**2} + B4 A43 C3^{**2} \\
& + B5 A52 C2^{**2} \\
& + B6 A62 C2^{**2} + B6 A63 C3^{**2} + B6 A64 C4^{**2} \\
& + B6 A65 C5^{**2} \\
& + B5 A53 C3^{**2} + B5 A54 C4^{**2} - 1/12) \\
& (+ B4 A43 A32 C2 + B5 A53 A32 C2 + B5 A54 A42 C2 \\
& + B5 A54 A43 C3 \\
& + B6 A63 A32 C2 + B6 A64 A42 C2 + B6 A64 A43 C3 \\
& + B6 A65 A52 C2 \\
& + B6 A65 A53 C3 + B6 A65 A54 C4 - 1/24) \\
& (+ B2 C2^{**4} + B3 C3^{**4} + B4 C4^{**4} + B5 C5^{**4} \\
& + B6 C6^{**4} - 1/5) \\
& (+ B3 C3^{**2} A32 C2 + B4 C4^{**2} A42 C2 + B4 C4^{**2} A43 C3 \\
& + B5 C5^{**2} A52 C2 + B5 C5^{**2} A53 C3 + B5 C5^{**2} A54 C4 \\
& + B6 C6^{**2} A62 C2 + B6 C6^{**2} A63 C3 + B6 C6^{**2} A64 C4 \\
& + B6 C6^{**2} A65 C5 - 1/10) \\
& (+ B3 C2^{**2} A32 C3 + B4 C2^{**2} A42 C4 + B4 C3^{**2} A43 C4 \\
& + B5 C2^{**2} A52 C5 + B5 C3^{**2} A53 C5 + B5 C4^{**2} A54 C5 \\
& + B6 C2^{**2} A62 C6 + B6 C3^{**2} A63 C6 + B6 C4^{**2} A64 C6 \\
& + B6 C5^{**2} A65 C6 - 1/15) \\
& (+ B4 C4 A43 A32 C2 + B5 C5 A53 A32 C2 + B5 C5 A54 A42 C2 \\
& + B5 C5 A54 A43 C3 \\
& + B6 C6 A63 A32 C2 + B6 C6 A64 A42 C2 + B6 C6 A64 A43 C2 \\
& + B6 C6 A65 A52 C2 + B6 C6 A65 A53 C3 + B6 C6 A65 A54 C4 \\
& - 1/30) \\
& (+ B3 A32^{**2} C2^{**2} + B4 A42^{**2} C2^{**2} + 2 B4 A42 C2 A43 C3 \\
& + B4 A43^{**2} C3^{**2} + B5 A52^{**2} C2^{**2} + B5 A53^{**2} C3^{**2} \\
& + B5 A54^{**2} C4^{**2} + 2 B5 A52 C2 A53 C3 \\
& + 2 B5 A52 C2 A54 C4 \\
& + 2 B5 A53 C3 A54 C4 \\
& + B6 A62^{**2} C2^{**2} + B6 A63^{**2} C3^{**2} + B6 A64^{**2} C4^{**2} \\
& + B6 A65^{**2} C5^{**2} \\
& + 2 B6 A62 C2 A63 C3 + 2 B6 A62 C2 A64 C4 \\
& + 2 B6 A62 C2 A65 C5 + 2 B6 A63 C3 A64 C4 \\
& + 2 B6 A63 C3 A65 C5 + 2 B6 A64 C4 A65 C5 \\
& - 1/20) \\
& (+ B3 A32 C2^{**3} + B4 A42 C2^{**3} + B4 A43 C3^{**3} \\
& + B5 A52 C2^{**3} \\
& + B5 A53 C3^{**3} + B5 A54 C4^{**3} \\
& + B6 A62 C2^{**3} + B6 A63 C3^{**3} + B6 A64 C4^{**3} \\
& + B6 A65 C5^{**3} - 1/20) \\
& (+ B4 A43 C3 A32 C2 + B5 A53 C3 A32 C2 + B5 A54 C4 A42 C2 \\
& + B5 A54 C4 A43 C3 \\
& + B6 A63 C3 A32 C2 + B6 A64 C4 A42 C2 + B6 A64 C4 A43 C3 \\
& + B6 A65 C5 A52 C2 + B6 A65 C5 A53 C3 + B6 A65 C5 A54 C4 \\
& - 1/40) \\
& (+ B4 A43 A32 C2^{**2} + B5 A53 A32 C2^{**2} + B5 A54 A42 C2^{**2}
\end{aligned}$$

```

+ B5 A54 A43 C3**2
+ B6 A63 A32 C2**2 + B6 A63 A42 C2**2 + B6 A64 A43 C3**2
+ B6 A65 A52 C2**2 + B6 A65 A53 C3**2 + B6 A65 A54 C4**2
- 1/60)
(+ B5 A54 A43 A32 C2
+ B6 A64 A43 A32 C2 + B6 A65 A53 A32 C2
+ B6 A65 A54 A42 C2 + B6 A65 A54 A43 C3
- 1/20)
)

```

Note: Groebner Basis not found after 1 000 000 ms.

5. 8 polynomials in 8 variables (Butcher, 1984)

```

$(BUTCHER, RUNGE-KUTTA, 20.1.1984 S = 3 PT = 4)
D = Q
R = D(B, C2, C3, A, B3, B2, A32, B1)
OPT = LIO P E10
(
(B1 + B2 + B3
- (A + B))
(B2 C2 + B3 C3
- (1/2 + 1/2 B + B**2 - A B))
(B2 C2**2 + B3 C3**2
- (A(1/3 + B**2) - 4/3 B - B**2 - B**3))
(B3 A32 C2
- (A(1/6 + 1/2 B + B**2) - 2/3 B - B**2 - B**3))
(B2 C2**3 + B3 C3**3
- (1/4 + 1/4 B + 5/2 B**2 + 3/2 B**3 + B**4
- A(B + B**3)))
(B3 C3 A32 C2
- (1/8 + 3/8 B + 7/4 B**2 + 3/2 B**3 + B**4
- A(1/2 B + 1/2 B**2 + B**3)))
(B3 A32 C2**2
- (1/12 + 1/12 B + 7/6 B**2 + 3/2 B**3 + B**4
- A(2/3 B + B**2 + B**3)))
(1/24 + 7/24 B + 13/12 B**2 + 3/2 B**3 + B**4
- A(1/3 B + B**2 + B**3))
)

```

Computing time for Groebner Basis on IBM 3081D:

1 382 210 ms

For univariate polynomials:

23 517 ms

Note: Infinitely many solutions, only one real solution satisfying additional inequalities.

*Series of examples occurring in the determination of truncated Laurent-series*

The examples have been communicated by Katsura (private communication).

1. 2 polynomials in 2 variables

\$(KATSURA, LAURENT SERIES CASE 1, 18. FEB. 1985)



---

```

D = Q
R = D(U0, U1)
OPT = IO PE5
(
(U0**2 - U0 + 2 U1**2)
(U0 + 2 U1 - 1)
)

```

Computing time for Groebner Basis on IBM 3081D:

17 ms

For univariate polynomials:

445 ms

*Note:* Finitely many solutions.

## 2. 3 polynomials in 3 variables

```

$(KATSURA, LAURENT SERIES CASE 2. 18. FEB. 1985)
D = Q
R = D(U0, U1, U2)
OPT = IO PE5
(
(U0**2 - U0 + 2 U1**2 + 2 U2**2)
(2 U0 U1 + 2 U1 U2 - U1)
(U0 + 2 U1 + 2 U2 - 1)
)

```

Computing time for Groebner Basis on IBM 3081D:

51 ms

For univariate polynomials:

1091 ms

*Note:* Finitely many solutions.

## 3. 4 polynomials in 4 variables

```

$(KATSURA, LAURENT SERIES CASE 3. 18. FEB. 1985)
D = Q
R = D(U0, U1, U2, U3)
OPT = IO PE5
(
(U0**2 - U0 + 2 U1**2 + 2 U2**2 + 2 U3**2)
(2 U0 U1 + 2 U1 U2 + 2 U2 U3 - U1)
(2 U0 U2 + U1**2 + 2 U1 U3 - U2)
(U0 + 2 U1 + 2 U2 + 2 U3 - 1)
)

```

Computing time for Groebner Basis on IBM 3081D:

4666 ms

For univariate polynomials:

15 612 ms

*Note:* Finitely many solutions.

## 4. 5 polynomials in 5 variables

```

$(KATSURA, LAURENT SERIES CASE 4, 18. FEB. 1985)
D = Q
R = D(U0, U1, U2, U3, U4)

```

```

OPT = IOG
(
(U0**2-U0+2 U1**2+2 U2**2+2 U3**2+2 U4**2)
(2 U0 U1+2 U1 U2+2 U2 U3+2 U3 U4-U1)
(2 U0 U2+U1**2+2 U1 U3+2 U2 U4-U2)
(2 U0 U3+2 U1 U2+2 U1 U4-U3)
(U0+2 U1+2 U2+2 U3+2 U4-1)
)

```

Computing time for Groebner Basis on IBM 3081D:

31 077 ms

*Note:* 16 Groebner Bases polynomials. Inverse graduated term ordering. Finitely many solutions (not yet known).

##### 5. 6 polynomials in 6 variables

```

$(KATSURA, LAURENT SERIES CASE 5, 18. FEB. 1985)
D = M 536870909 (= 2**29-3)
R = D(U0, U1, U2, U3, U4, U5)
OPT = IOG
(
(U0**2-U0+2 U1**2+2 U2**2+2 U3**2+2 U4**2
+2 U5**2)
(2 U0 U1+2 U1 U2+2 U2 U3+2 U3 U4+2 U4 U5-U1)
(2 U0 U2+U1**2+2 U1 U3+2 U2 U4+2 U3 U5-U2)
(2 U0 U3+2 U1 U2+2 U1 U4+2 U2 U5-U3)
(2 U0 U4+2 U1 U3+2 U1 U5+U2**2-U4)
(U0+2 U1+2 U2+2 U3+2 U4+2 U5-1)
)

```

Computing time for Groebner Basis on IBM 3081D:

87 142 ms

*Note:* 32 Groebner Bases polynomials. Inverse graduated term ordering, computation modulo a large prime.

##### *Some special examples*

###### 1. Example communicated by Gerdt *et al.* (1985).

###### 13 polynomials in 7 variables

```

$(GERDT, 10.10.84)
D = Q
R = D(L1, L2, L3, L4, L5, L6, L7)
OPT = OIL PE10
(
(L1(L4-1/2 L5+L6))
((2/7 L1**2-L4) (-10 L1+5 L2-L3))
((2/7 L1**2-L4) (3 L4-L5+L6))
((-2 L1**2+L1 L2+2 L1 L3-L2**2-7 L5+21 L6)
(-3 L1+2 L2)+21(7 L7-2 L1 L4+3/7 L1**3)

```

```

((-2 L1**2+L1 L2+2 L1 L3-L2**2-7 L5+21 L6)
(2 L4-2 L5)+(7 L7-2 L1 L4+3/7 L1**3)
(-45 L1+15 L2-3 L3))
(2(-2 L1**2+L1 L2+2 L1 L3-L2**2-7 L5+21 L6)
L7+(7 L7-2 L1 L4+3/7 L1**3)
(12 L4-3 L5+2 L6))
((L1 (5 L1-3 L2+L3))
(2 L2-L1)
+7(L1(2 L6-4 L4)))
((L1(5 L1-3 L2+L3))
L3+7(L1(2 L6-4 L4))
)
((L1(5 L1-3 L2+L3))
(-2 L4-2 L5)+(L1(2 L6-4 L4))
(2 L2-8 L1)+84 1/2 L1 L7
)
((L1(5 L1-3 L2+L3))
(8/3 L5+6 L6)+(L1 (2 L6-4 L4))
(11 L1-17/3 L2+5/3 L3)-168 1/2 L1 L7
)
(15 L7
(L1(5 L1-3 L2+L3))
+(L1(2 L6-4 L4)) (5 L4-2 L5)
+1/2 L1 L7 (-120 L1+30 L2-6 L3))
(-3(L1(5 L1-3 L2+L3))
L7+(L1 (2 L6-4 L4))
(-1/2 L4+1/4 L5-1/2 L6)+1/2 L1 L7
(24 L1-6 L2))
(3(L1(2 L6-4 L4))
L7+1/2 L1 L7
(40 L4-8 L5+4 L6))
)

```

Computing time for Groebner Basis on IBM 3081D:

55430 ms

*Note:* Infinitely many solutions.

## 2. Example from Raksanyi & Walter (1983).

Problem occurring in systems theory with rational function coefficients.

4 polynomials in 4 variables

\$(RAKSANYI 1, 1983 RATIONAL FUNCTIONS.)

D = F(A1, A2, A3, A4)

R = (X1, X2, X3, X4)

OPT = OIL

(

(X4-(A4-A2))

(X1+X2+X3+X4-(A1+A3+A4))

```
(X1 X3 + X1 X4 + X2 X3 + X3 X4 - (A1 A4 + A1 A3 + A3 A4))
(X1 X3 X4 - (A1 A3 A4))
)
```

Computing time for Groebner Basis on IBM 3081D:

1507 ms

*Note:* Solvable for transcendental  $A_1, A_2, A_3, A_4$ . Moreover solvable for numerical values of  $A_1, A_2, A_3, A_4$  if  $A_2 \neq A_4$ .

3. Example arising in a general economic equilibrium model (Shoven, 1983) and discussed in Rose *et al.* (1984).

```
$(ROSE, GENERAL EQUILIBRIUM MODEL, 1984)
D = Q
R = D(U3, U4, A46)
OPT = IOG
(
(U4**4 - 20/7 A46**2)
(A46**2 U3**4 + 7/10 A46 U3**4 + 7/48 U3**4 - 50/27
A46**2 - 35/27 A46 - 49/216)
(A46**5 U4**3 + 7/5 A46**4 U4**3 + 609/1000 A46**3
U4**3 + 49/1250 A46**2 U4**3 - 27391/800000 A46 U4**3
- 1029/160000 U4**3 + 3/7 A46**5 U3 U4**2 + 3/5 A46**6
U3 U4**2 + 63/200 A46**3 U3 U4**2 + 147/2000 A46**2
U3 U4**2 + 4137/800000 A46 U3 U4**2 - 7/20 A46**4
U3**2 U4 - 77/125 A46**3 U3**2 U4 - 23863/60000 A46**2
U3**2 U4 - 1078/9375 A46 U3**2 U4 - 24353/1920000
U3**2 U4 - 3/20 A46**4 U3**3 - 21/100 A46**3 U3**3
- 91/800 A46**2 U3**3 - 5887/200000 A46 U3**3
- 343/128000 U3**3)
)
```

Computing time for Groebner Basis on IBM 3081D:

193 197 ms

*Note:* It was not possible to compute a Groebner Basis with respect to the inverse lexicographical term ordering. Moreover, the univariate polynomials for  $U_3$  and for  $U_4$  could not be computed from the Groebner Basis in the inverse graduated term ordering.

The univariate minimal polynomials (each having degree 30 and 8 real roots) were computed only for  $A_{46}$ ,  $P_4 = U_4^{**4}$  and  $P_3 = U_3^{**4}$ . Using the inverse lexicographical term ordering these polynomials occurred in the Groebner Basis.

This gives also a finite inclusion for the real zeros. It turned out, that there are 8 tuples of real solutions of the system of algebraic equations in  $A_{46}, P_3, P_4$ .

4. Example from Gonnet *et al.* (1983).

19 polynomials in 17 variables

```
$(UNIVERSITY OF WATERLOO, 19.03.1984)
D = Q
R = D(A0, A2, A3, A4, A5, B0, B1, B2, B3, B4, B5, C0, C1, C2, C3, C4, C5)
OPT = OIL PE10
(
```

```

(A4 B4)
(A5 B1 + B5 + A4 B3 + A3 B4)
(A2 B2)
(A5 B5)
(A0 B2 + B2 + A4 B2 + A2 B4 + C2 + A2 B0 + A2 B1)
(A0 B0 + A0 B1 + A0 B4 + A3 B2 + B0 + B1 + B4 + A4 B0
+ A4 B1
+ A2 B5 + A4 B4 + C1 + C4 + A5 B2 + A2 B3 + C0)
(A3 B0 + A0 B3 + A0 B5 + A5 B0 + B3 + B5 + A5 B4 + A4 B3 +
A4 B5 + A3 B4 + A5 B1 + A3 B1 + C3 + C5 - 1)
(A5 B3 + A5 B5 + A3 B5 + A3 B3)
(A5 B3 + 2 A5 B5 + A3 B5)
(A0 B5 + A5 B0 + A3 B4 + 2 A5 B4 + A5 B1 + B5 + A4 B3
+ 2 A4 B5 + C5)
(A4 B0 + 2 A4 B4 + A2 B5 + B4 + A4 B1 + A5 B2 + A0 B4
+ C4)
(A2 B4 + A4 B2)
(A4 B5 + A5 B4)
(2 A3 B3 + A5 B3 + A3 B5)
(C3 + A0 B3 + 2 B3 + B5 + A4 B3 + A3 B0 + 2 A3 B1 +
A5 B1 + A3 B4)
(C1 + A0 B1 + 2 B1 + A4 B1 + A2 B3 + B0 + A3 B2 + B4)
(A2 B1 + B2)
(A5 B3 + A3 B5)
(B4 + A4 B1)
)

```

Computing time for Groebner Basis on IBM 3081D:

9529 ms

*Note:* Infinitely many solutions. With this method it is possible to find *all* three solution subsets, not only the one communicated by Gonnet *et al.* (1983) but see also Gebauer & Kredel (1984b).

## 5. Examples from Trinks (1978).

6 polynomials in 6 variables

```

$(TRINKS 1, IDEAL A. 09.12.1983)
D = Q
R = D(B, S, T, Z, P, W)
OPT = I
(
(+45 P + 35 S - 165 B - 36)
(+35 P + 40 Z + 25 T - 27 S)
(+15 W + 25 P S + 30 Z - 18 T - 165 B**2)
(-9 W + 15 P T + 20 Z S)
(W P + 2 Z T - 11 B**3)
(99 W - 11 S B + 3 B**2)
)

```

Computing time for Groebner Basis on IBM 3081D: 138 425 ms  
 For univariate polynomials: 983 743 ms

*Note:* Finitely many solutions.

7 polynomials in 6 variables

\$(TRINKS 2, IDEAL P = A + F7 R. 10.12.1983)

D = Q

R = D (B, S, T, Z, P, W)

OPT = IL

(  
 + 45 P + 35 S - 165 B - 36,  
 + 35 P + 40 Z + 25 T - 27 S,  
 + 15 W + 25 P S + 30 Z - 18 T - 165 B\*\*2,  
 - 9 W + 15 P T + 20 Z S,  
 W P + 2 Z T - 11 B\*\*3,  
 99 W - 11 S B + 3 B\*\*2,  
 B\*\*2 + 33/50 B + 2673/10000  
 )

Computing time for Groebner Basis on IBM 3081D: 888 ms  
 For univariate polynomials: 2164 ms

*Note:* Finitely many solutions.

#### 4. Conclusions About the Applicability of the Method

##### *Summary of computing times*

Computing times for computing Groebner Bases with different variable and term orderings for one fixed example showing the strong dependency on variable and term ordering:

| variable<br>ordering | time<br>OPT = L | time<br>OPT = G |
|----------------------|-----------------|-----------------|
| BSTZPW               | 1950            | 10 920          |
| SBTZPW               | 27 890          | 16 370          |
| STBZPW               | 110 270         | 18 740          |
| STZPBW               | 115 350         | 16 370          |
| STZPWB               | 127 030         | 20 810          |
| SZPWBT               | 67 680          | 19 780          |
| PWBTSZ               | 103 160         | 20 610          |
| ZWBSTP               | > 3 600 000     | 32 310          |
| TZPWBS               | tfc             | 50 500          |
| ZPWBST               | > 3 600 000     | 39 320          |
| PWBSTZ               | 37 180          | 20 880          |
| WBSTZP               | 34 980          | 10 810          |

Computing time in milliseconds on IBM 370/168.

'tfc' means 'too few cells reclaimed', i.e. not enough storage available. These results motivate the heuristics for finding an "optimal" variable ordering as described above.

Times for the computation of Groebner Bases in the above examples.

| Example    | Variables/<br>Polynomials | total degree<br>Input/Output | Options<br>Domain | computing<br>time in ms |
|------------|---------------------------|------------------------------|-------------------|-------------------------|
| Hairer 1   | 8/6                       | 3/3                          | IOL Q             | 482                     |
| Hairer 2   | 13/11                     | 4/5                          | IOL Q             | 415 558                 |
| Hairer 3   | 14/16                     | 5/0                          | IOG Q             | 99 685                  |
| Hairer 4   | 20/14                     | 5/?                          | IOL Q             | ?                       |
| Butcher    | 8/8                       | 4/7                          | IOL Q             | 1 382 210               |
| Katsura 1  | 2/2                       | 2/2                          | IOL Q             | 17                      |
| Katsura 2  | 3/3                       | 2/3                          | IOL Q             | 51                      |
| Katsura 3  | 4/4                       | 2/7                          | IOL Q             | 4666                    |
| Katsura 4  | 5/5                       | 2/5                          | IOG Q             | 31 077                  |
| Katsura 5  | 6/6                       | 2/6                          | IOG M             | 87 142                  |
| Gerdt      | 7/13                      | 3/6                          | IOL Q             | 55 430                  |
| Raksanyi 1 | 4/4                       | 3/3                          | IOL F             | 1507                    |
| Raksanyi 2 | 4/4                       | 3/5                          | IOL F             | 1365                    |
| Rose       | 3/3                       | 8/10                         | IOG Q             | 193 197                 |
| Gonnet     | 17/19                     | 2/2                          | IOL Q             | 9529                    |
| Trinks 1   | 6/6                       | 3/10                         | I L Q             | 138 425                 |
| Trinks 2   | 6/7                       | 3/2                          | I L Q             | 888                     |

Computing time in milliseconds on IBM 3081D.

*From these computing times one may draw the following conclusions:*

- Computing time seems to grow exponentially in the number of variables.
- The construction of Groebner Bases using the inverse lexicographical term ordering fails in some examples because of computing time.
- Using the inverse graduated term ordering instead of the inverse lexicographical ordering, one often can compute Groebner Bases for problems in more variables. But then the construction of the univariate polynomials needs more computing time.
- Doing all computations modulo a large prime, Groebner Bases can be found for more variables. However, the subsequent construction of the Groebner Basis over  $\mathbb{Q}$  and the construction of the univariate polynomials is an open problem, see for instance Trinks (1984) and Ebert (1983).
- For some systems with special structure (such as the ones occurring in the construction of Runge-Kutta methods) it is possible to compute Groebner Bases for even 14 variables.

## 5. Directions for Future Research

- A quickening of this method of solving systems of algebraic equations must concentrate on speeding up the computation of Groebner Bases. Much research is still needed.
- It should be possible to make the construction of the univariate polynomials faster by taking into account that series of sets of linear equations have to be solved until one is found that is solvable.
- The method of ideal decomposition should be extended to treat ideals having infinitely many solutions.

- The combination of the real roots of the univariate polynomials should be done automatically by a program in case a mathematical criterion for selecting "valid" combinations is available.
- Programs for the computation of the complex roots of univariate polynomials should also be implemented.
- During the computation of Groebner Bases the (multivariate) factorisation of the intermediate polynomials should be considered.

Complete listings of the above (and other) examples are available from the authors. The programs are available on request.

### References

- Buchberger, B. (1965). Ein Algorithmus zum Auffinden der Basiselemente des Restklassenringes nach einem nulldimensionalen Polynomideal. PhD Thesis, University of Innsbruck.
- Buchberger, B. (1970). Ein algorithmisches Kriterium fuer die Loesbarkeit eines algebraischen Gleichungssystems. *Aequ. Math.* **4**, 374–383.
- Buchberger, B. (1985). Groebner Bases: An Algorithmic Method in Polynomial Ideal Theory. In: (Bose, N. K. ed.) *Progress, directions and open problems in multidimensional systems theory*. pp. 184–232. Dordrecht: Reidel.
- Butcher, C. J. (1984). An application of the Runge Kutta space. *BIT Computer Science Numerical Mathematics* **24**, 425–440.
- Collins, G. E., Loos, R. G. (1980). SAC-2—Symbolic and Algebraic Computation version 2, a computer algebra system, ALDES—ALgorithm DESCRIPTION language. *ACM SIGSAM Bulletin* **14**, No. 2.
- Collins, G. E., Loos, R. G. (1982). Real zeros of polynomials. In: (Buchberger, B., Collins, G., Loos, R. eds.) *Computer algebra—symbolic and algebraic computation*. Vienna: Springer.
- Ebert, G. L. (1983). Some comments on the modular approach to Groebner bases. *ACM SIGSAM Bulletin* **17**, No. 2, 28–32.
- Gebauer, R., Kredel, H. (1983a). *Common distributive polynomial system*. Technical Report, Institut für Angewandte Mathematik, Universität Heidelberg.
- Gebauer, R., Kredel, H. (1983b). *Buchberger algorithm system*. Technical Report, Institut für Angewandte Mathematik, Universität Heidelberg. (see also *ACM SIGSAM Bulletin* **18**, No. 1, 19).
- Gebauer, R., Kredel, H. (1984a). *Real solution system for algebraic equations*. Technical Report, Institut für Angewandte Mathematik, Universität Heidelberg.
- Gebauer, R., Kredel, H. (1984b). A note to: "Solution of a general system of algebraic equations". *ACM SIGSAM Bulletin* **18**, No. 3, 5–6.
- Gerdt, V. P., Shavachka, A. B., Zharkov, A. (1985). Computer algebra application for classification of integral non-linear evolution equations. *J. Symbolic Computation* **1**, 101–107.
- Gonnet, G. H., Char, B. W., Geddes, K. O. (1983). Solution of a general system of equations. *ACM SIGSAM Bulletin* **17**, No. 3 & 4, 48–49.
- Hearn, A. C. (1985). *REDUCE user's manual*, version 3.2. Santa Monica: The Rand Corporation.
- Henrici, P. (1961). *Discrete variable methods in ordinary differential equations*. New York: Wiley.
- Kaltofen, E. (1982). Factorisation of polynomials. In: (Buchberger, B., Collins, G., Loos, R.) *Computer algebra—symbolic and algebraic computation*. Vienna: Springer.
- Raksanyi, A., Walter, E. (1983). Une application du calcul formel: le test des propriétés structurelles des modèles d'état. *Proceedings of Conference "Algorithmique, Calcul Formel, Arithmétique"*. Publications de l'Université de Saint-Etienne.
- Rose, M., Gebauer, R., Kredel, H., Kungl, H. (1984). Exemplifying the workability of the Buchberger algorithm for computation of general equilibrium by a model published in Shoven (1983). Departments of Economics and Applied Mathematics, University of Heidelberg.
- Schrader, R. (1976). *Zur konstruktiven Idealtheorie (Diplomarbeit)*. Mathematisches Institut II, Universität Karlsruhe.
- Shoven, J. B. (1983). *Applied general equilibrium tax modelling*. IMF Staff Papers, pp. 394–419.
- Stoer, J., Bulirsch, R. (1979). *Introduction to numerical analysis*. New York: Springer.
- Trinks, W. L. (1978). Ueber Buchbergers Verfahren Systeme algebraischer Gleichungen zu loesen. *J. Number Theor.* **10**, 475–488.
- Trinks, W. L. (1984). On improving approximate results of Buchberger's algorithm by Newton's method. *ACM SIGSAM Bulletin* **18**, No. 3, 5–6.
- Wrinkler, F., Buchberger, B., Lichtenberger, F., Rolletschek, H. (1985). An algorithm for constructing canonical bases (Groebner bases) of polynomial ideals. *ACM/TOMS* **11**, 66–78.